ประพนธ์ อัศวภาณุวัฒน์

SOFT MAIL

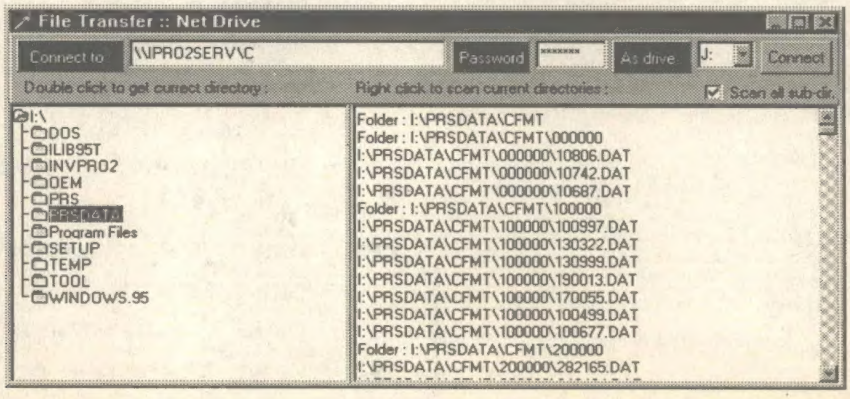
Communications : Files Transfer (ตอนที่ 1)

คุยกันเรื่องการสื่อสาร ระหว่างเครื่องคอมพิวเตอร์ กล่าวถึงเรื่องการโอนย้ายแฟ้มข้อมูล ระหว่างเครื่องผ่านระบบเครือข่าย

สวัสดีครับ ในฉบับนี้เราจะคุยกันเรื่องการสื่อสารระหว่างเครื่องคอมพิวเตอร์ ในตอนแรกนี้ผมจะขอกล่าวถึงเรื่องการโอนย้ายแฟ้มข้อมูลระหว่างเครื่องสามารถช่วยเราโอนย้ายแฟ้มข้อมูลจากเครื่องหนึ่งไปยังอีกเครื่องหนึ่งผ่านระบบเครือข่ายได้ โดยผมได้เตรียมกรรมวิธีไว้ 2 วิธี วิธีแรกเป็นการใช้ประโยชน์จากระบบเครือข่ายของระบบวินโดวส์เองคือ การ Share ดิสก์ไดรฟ์ (Disk Drive) ผมขอเรียกวิธีนี้ว่า "Net Drive" และวิธีที่สองเป็นการใช้ FTP ในการโอนย้ายข้อมูลทั้งสองวิธีมีวิธีการใช้งานแตกต่างกันอย่างไรติดตามได้ในเรื่องเลยครับ

File Transfer : Net Drive

Net Drive เป็นกรรมวิธีการโอนย้ายข้อมูลผ่านระบบเครือข่าย ภายใต้วิธีการของระบบวินโดวส์คือการ Share ดิสก์ไดรฟ์ กล่าวคือ เครื่องที่ทำหน้าที่เป็น Host จะต้อง Share ดิสก์ไดรฟ์ของตนเองให้เป็น Network Drive โดยกำหนดชื่อที่อ้างถึง (Alias) วิธีการเข้าถึง (Access Mode) และความสามารถในการเข้าสู่ระบบ (User Permission) สำหรับเครื่องที่ต้องการคัดลอกแฟ้มข้อมูลก็เพียงแต่ Connect ไดรฟ์ไปยัง Network Drive ที่ต้องการ แล้วตั้งชื่อไดรฟ์และป้อนรหัสผ่าน ก็สามารถใช้งานไดรฟ์ที่ Connect จากเครื่อง Host (ภายใต้เงื่อนไขของ Host) เสมือนไดรฟ์ของตนเอง ซึ่งเป็นวิธีปกติที่ผู้ใช้ระบบเครือข่ายทราบและสามารถปฏิบัติได้อยู่แล้ว ข้อจำกัดของระบบนี้คือผู้ใช้ส่วนใหญ่จะไม่ค่อยนิยม Disconnect ไดรฟ์ในทันทีที่ใช้งานเสร็จ (ส่วนใหญ่จะค้างไว้จนปิดเครื่อง และร้ายกว่านั้นถ้าเผลอใช้ Option : Reconnect at logon แล้ว จะเป็นการ Connect ไว้ตลอดที่มีการใช้เครื่องนี้) ปัญหาคือถ้าเครื่อง Host ไม่ใช่ระบบ Windows NT Server แล้ว ผู้ใช้ตั้งแต่คนที่ 11 จะไม่สามารถเข้าถึง Host ได้ เนื่องจากระบบวินโดวส์โดยทั่วไปจะจำกัดการติดต่อพร้อม ๆ กันไว้เพียงแค่ 10 Connection เท่านั้น และการต่อ Network Drive ทิ้งไว้อาจจะเป็นการเพิ่มภาระให้กับระบบเครือข่าย เนื่องจากระบบจะส่ง Notify Message กันตลอดเวลา



รูปที่ 1

โปรแกรม NetDrive

สำหรับโปรแกรม NetDrive แสดงอยู่ในรูปที่ 1 ใช้ระบบการ Connect ดิสก์ไดรฟ์เช่นกัน แต่พยายามแก้ไขปัญหา Connection ของระบบ โดยโปรแกรมจะทำการเชื่อมต่อไดรฟ์ (Map Drive) เมื่อโปรแกรมเริ่มทำงาน และทำการ Disconnect ไดรฟ์ทันทีที่ยุติการรัน โปรแกรม Net Drive จะแสดงแฟ้มข้อมูลต่าง ๆ ใน Network Drive เมื่อมีการคลิกขวาที่ Folder ตัวโปรแกรมแสดงอยู่ในลิสติงที่ 1.1 และส่วน Components ต่าง ๆ แสดงอยู่ในลิสติงที่ 1.2 มีรายละเอียดที่สำคัญดังนี้

กิจกรรม GetConnectionError(errId : integer); เป็นกิจกรรมสำหรับแสดง Error Message ที่เกิดขึ้นจากการพยายามติดต่อกับ Network Drive

กิจกรรม GetDisConnectError(errId : integer); เป็นกิจกรรมสำหรับแสดง Error Message ที่เกิดขึ้นจากการพยายามยกเลิกการติดต่อกับ Network Drive

กิจกรรม Connect2RemoteSrvr(rmtName, pswd, localDrv : string) : boolean; กิจกรรมติดต่อ Network Drive มี rmtName เป็นชื่อเครื่อง Host ที่ทำการ Share ไดรฟ์และตามด้วยชื่อ Alias ของ Network device ที่ Share เช่น เครื่องชื่อ 'COMPUTER' และ Share ไดรฟ์ C จะได้เป็น "\\COMPUTER\C", pswd เป็นรหัสผ่าน และ localDrv เป็นชื่อไดรฟ์ที่ต้องการใช้เรียก Network Drive ถ้าสามารถ Connect ได้จะ Return ค่าเป็น TRUE

ในกิจกรรม Connect2RemoteSrvr จะเรียกใช้ฟังก์ชัน WNetAddConnection ซึ่งเป็นฟังก์ชันในระบบ Win32 ใช้ทำการติดต่อกับ Network Drive มีวิธีการใช้คือ

```
DWORD WNetAddConnection(
  LPTSTR lpszRemoteName, // address of network device name
  LPTSTR lpszPassword, // address of password
  LPTSTR lpszLocalName  // address of local device name
  );
```

โดยมีตัวแปร : lpszRemoteName ระบุ Address ของ Network device name

lpszPassword ระบุ Address ของรหัสผ่าน

lpszLocalName ระบุชื่อไดรฟ์ที่ต้องการใช้เป็น Local Drive

return ค่า : เป็น "NO_ERROR" ถ้าติดต่อสำเร็จ

กิจกรรม DisconnectDrive(netDrv : string); เป็นกิจกรรมสำหรับยกเลิกการติดต่อกับ Network Drive

กิจกรรม DisconnectDrive จะเรียกใช้ฟังก์ชัน WNetCancel Connection เพื่อยุติการติดต่อ มีวิธีการใช้คือ

```
DWORD WNetCancelConnection(
  LPTSTR lpszName, // address of resource name to be
  disconnected
  BOOL fForce // flag for unconditional disconnect
  );
```

มีตัวแปร : lpszName ระบุ Address ของชื่อไดรฟ์ที่ต้องการยกเลิก

: fForce เป็น Flag ถ้าระบุ 'TRUE' เป็นการ Force ให้ยกเลิกการติดต่อถึงแม้มีการเปิดแฟ้มข้อมูลค้างไว้

กิจกรรม GetAllFileLists(curFolder : string; fileList : TStringList); เป็นกิจกรรมชนิด Recursive Method สำหรับค้นหารายชื่อแฟ้มข้อมูลที่ระบุใน curFolder และใส่รายชื่อที่ค้นพบไว้ใน fileList

กิจกรรม GetLastDrive : string; เป็นกิจกรรมค้นหาไดรฟ์ตัวสุดท้าย ที่มีใช้อยู่ในระบบ

กิจกรรม GetLastDrive จะเรียกใช้ฟังก์ชัน Win32 คือ GetLogicalDriveStrings ซึ่งเป็นกิจกรรมใช้ Retrive รายชื่อไดรฟ์ที่มีอยู่ในระบบ มีวิธีการใช้คือ

```
DWORD GetLogicalDriveStrings(
  DWORD nBufferLength, // size of buffer
  LPTSTR lpBuffer // address of buffer for drive strings
  );
```

มีตัวแปร: nBufferLength ระบุขนาดของ buffer 'lpBuffer'

: lpBuffer ระบุ Address ของ Buffer ที่ใช้รับรายชื่อไดรฟ์ที่มีอยู่ในระบบ

กิจกรรม AddDrv2DrvList; เป็นกิจกรรมสำหรับแสดงรายชื่อไดรฟ์ที่สามารถใช้งานได้ใน Combo Box

กิจกรรม RemoveAllNetDrive; เป็นกิจกรรมสำหรับยกเลิกการติดต่อกับ Network Drive โดยเรียกใช้ฟังก์ชัน Win32 คือ GetDriveType เพื่อตรวจสอบชนิดไดรฟ์ ถ้าเป็นชนิด "DRIVE_REMOTE" ก็จะยกเลิกการติดต่อ มีวิธีการใช้คือ

```
UINT GetDriveType(
  LPCTSTR lpRootPathName      // address of root path
  );
```

มีตัวแปร : lpRootPathName เป็น Pointer ไปยัง Address ของไดรฟ์ที่ต้องการทราบ

Return ค่า: เป็นชนิดไดรฟ์คือ

0 : ไม่สามารถตรวจสอบได้

1 : ไม่พบไดรฟ์ที่ระบุ

DRIVE_REMOVABLE : ไดรฟ์ชนิด Removabled (สามารถถอดได้)

DRIVE_FIXED : ไดรฟ์ชนิด Fixed Drive เช่น Hard Disk

DRIVE_REMOTE : ไดรฟ์ชนิด Remote network drive

DRIVE_CDROM : CD-ROM ไดรฟ์

DRIVE_RAMDISK : RAM ไดรฟ์

ลิสติงที่ 1.1

```
00001:  // Listing 1.1 : Net Drive
00002:  //
00003:  unit uNetMain;
00004:  {$H-}
00005:  interface
00006:  uses
00007:    Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,
00008:    StdCtrls, ComCtrls, ExtCtrls, Grids, Outline, DirOutln;
00009:  type
00010:    TFNetDrv = class(TForm)
00011:      Panel5: TPanel;
00012:      Edit2: TEdit;
00013:      Panel3: TPanel;
00014:      Panel4: TPanel;
00015:      DirectoryOutline1: TDirectoryOutline;
00016:      CBDrvList: TComboBox;
00017:      Button2: TButton;
00018:      RichEdit1: TRichEdit;
00019:      Splitter1: TSplitter;
00020:      Panel1: TPanel;
00021:      Edit1: TEdit;
00022:      Label1: TLabel;
00023:      Label2: TLabel;
00024:      CheckBox1: TCheckBox;
00025:      procedure FormCreate(Sender: TObject);
00026:      procedure FormCloseQuery(Sender: TObject; var CanClose: Boolean);
00027:      procedure Button2Click(Sender: TObject);
00028:      procedure DirectoryOutline1MouseDown(Sender: TObject;
00029:        Button: TMouseButton; Shift: TShiftState; X, Y: Integer);
00030:      procedure Splitter1Moved(Sender: TObject);
00031:    private
00032:      { Private declarations }
00033:        fileList : TStringList;
00034:      procedure InitVar;
00035:      procedure ReleaseVar;
00036:      procedure GetConnectionError( errId : integer );
00037:      procedure GetDisConnectError( errId : integer );
00038:      function  Connect2RemoteSrvr( rmtName, pswd, localDrv : string ) : boolean;
00039:      procedure DisconnectDrive( netDrv : string );
00040:      procedure GetAllFileLists( curFolder : string; fileList : TStringList );
00041:      function  GetLastDrive : string;
00042:      procedure AddDrv2DrvList;
00043:      procedure RemoveAllNetDrive;
00044:    public
00045:      { Public declarations }
00046:    end;
00047:  var
00048:    FNetDrv: TFNetDrv;
00049:  implementation
00050:  {$R *.DFM}
00051:  //- Initialized section ---------------------------------
00052:  procedure TFNetDrv.InitVar;
00053:  begin
00054:    fileList:=TStringList.Create;
00055:    AddDrv2DrvList;
00056:  end;
00057:  procedure TFNetDrv.ReleaseVar;
00058:  begin
00059:    RemoveAllNetDrive;
00060:    fileList.free;
00061:  end;
00062:  //- Net Managerment Section -----------------------------
00063:  procedure TFNetDrv.GetConnectionError( errId : integer );
00064:  var s : string;
00065:  begin
00066:    case errId of
00067:      ERROR_ACCESS_DENIED : s:='Accesss denied';
00068:      ERROR_ALREADY_ASSIGNED : s:='Local drive already connected';
00069:      ERROR_BAD_DEV_TYPE : s:='Device Type and resource type not match';
00070:      ERROR_BAD_DEVICE : s:='Local drive name is invalid';
00071:      ERROR_BAD_NET_NAME : s:='Network device is invalid';
00072:      ERROR_BAD_PROFILE : s:='User profile is incorrect format';
00073:      ERROR_CANNOT_OPEN_PROFILE : s:='The system is unable to open the user'+
```

→

ลิสติงที่ 1.1 (ต่อ)

```
00074:          ' profile to process persistent connections';
00075:        ERROR_DEVICE_ALREADY_REMEMBERED : s:='An entry for the device specified'+
00076:          ' in local drive is already in the user profile';
00077:        ERROR_EXTENDED_ERROR : s:='A network-spcific error occurred';
00078:        ERROR_INVALID_PASSWORD : s:='The specified password is invalid';
00079:        ERROR_NO_NET_OR_BAD_PATH : s:='The operation cannot be performed because'+
00080:          'either a network component is not started or the specified name'+
00081:          'can not be used.';
00082:        ERROR_NO_NETWORK : s:='The network is not present';
00083:        else s:='Unknown error, error code = '+IntToStr( errId );
00084:      end;
00085:      ShowMessage( s );
00086:  end;
00087:  procedure TFNetDrv.GetDisConnectError( errId : integer );
00088:  var s : string;
00089:  begin
00090:    case errId of
00091:      ERROR_BAD_PROFILE : s:='The user profile is in an incorrect format';
00092:      ERROR_CANNOT_OPEN_PROFILE : s:='The system is unable to open the user'+
00093:        'profile to process persistent connections';
00094:      ERROR_DEVICE_IN_USE : s:='The device is in use by an active process and'+
00095:        'cannot be disconnected';
00096:      ERROR_EXTENDED_ERROR : s:='A network-specific error occurred';
00097:      ERROR_NOT_CONNECTED : s:='The name specified by the "netdrv" parameter'+
00098:        'is not a redirected device, or the system is not currently connected'+
00099:        ' to the device specified by the parameter';
00100:      ERROR_OPEN_FILES : s:='There are open files, and [fForce] parameter'+
00101:        'is FALSE';
00102:    end;
00103:  end;
00104:  function TFNetDrv.Connect2RemoteSrvr( rmtName, pswd, localDrv : string ) : boolean;
00105:  var ret : longint;
00106:      rmtNameP, pswdP, localDrvP : pchar;
00107:  begin
00108:    rmtNameP:=@rmtName[1]; pswdP:=@pswd[1]; localDrvP:=@localDrv[1];
00109:    rmtName:=rmtName+#0; pswd:=pswd+#0; localDrv:=localDrv+#0;
00110:    ret:=WNetAddConnection( rmtNameP, pswdP, localDrvP );
00111:    if ( ret<>NO_ERROR ) then
00112:    begin
00113:      GetConnectionError( ret ); result:=false;
00114:    end else result:=true;
00115:  end;
00116:  procedure TFNetDrv.DisconnectDrive( netDrv : string );
00117:  var ret : longint; netDrvP : pchar;
00118:  begin
00119:    netDrvP:=@netDrv[1]; netDrv:=netDrv+#0;
00120:    ret:=WNetCancelConnection( netDrvP, TRUE );
00121:    if ( ret<>NO_ERROR ) then GetDisConnectError( ret );
00122:  end;
00123:  //-----------------------------------------
00124:  procedure TFNetDrv.GetAllFileLists( curFolder : string; fileList : TStringList );
00125:  var found : integer; searchRec : TSearchRec; tmp : string;
00126:      fileExt, filename : string[80];
00127:  begin
00128:    if ( fileList=nil ) then exit;
00129:    fileExt:='\*.*';
00130:    found := FindFirst( curFolder+fileExt, faAnyFile, searchRec );
00131:    if ( found<>0 ) then begin FindClose( SearchRec ); exit; end;
00132:    while found = 0 do
00133:    begin
00134:      if ( searchRec.attr=$00000010 ) then
00135:      begin
00136:        tmp:=searchRec.name;
00137:        if ( tmp<>'.' ) and ( tmp<>'..' ) then
00138:        begin // recursed folder
00139:          tmp:=curFolder+'\'+tmp;
00140:          fileList.add( 'Folder : '+tmp );
00141:          if ( CheckBox1.checked ) then GetAllFileLists( tmp, fileList );
00142:        end;
00143:      end else
00144:      begin
00145:        if ( searchRec.attr<>$00000008 ) then
00146:        begin
```

→

ลิสติงที่ 1.1 (ต่อ)

```
00147:           fileName:=curFolder+'\'+searchRec.name;
00148:           fileList.add( fileName );
00149:         end;
00150:       end;
00151:       found:=FindNext( searchRec );
00152:     end;
00153:     FindClose( searchRec );
00154:   end;
00155:   function TFNetDrv.GetLastDrive : string;
00156:   var s,lastDrv : String; sP : pChar; ret : integer;
00157:   begin
00158:     sP:=@s[1]; fillchar( s, sizeof( String ), #0 );
00159:     ret:=GetLogicalDriveStrings( sizeof( String ), sP );
00160:     s[0]:=chr(ret); lastDrv:='';
00161:     repeat
00162:       lastDrv:=copy( s,1,pos( #0,s )-1 ); delete( s,1, pos( #0,s ) );
00163:     until ( s='' );
00164:     result:=lastDrv;
00165:   end;
00166:   procedure TFNetDrv.AddDrv2DrvList;
00167:   var lastDrv : string; c : integer;
00168:   begin
00169:     CBDrvList.items.clear;
00170:     lastDrv:=GetLastDrive;
00171:     for c:=ord( lastDrv[1] )+1 to ord( 'z' ) do
00172:       CBDrvList.items.add( UpCase(chr( c ))+':' );
00173:     CBDrvList.text:=CBDrvList.items[0];
00174:   end;
00175:   procedure TFNetDrv.RemoveAllNetDrive;
00176:   var lastDrv : string; c : integer; lastDrvP : pchar; ret : word;
00177:   begin
00178:     lastDrvP:=@lastDrv[1];
00179:     repeat
00180:       lastDrv:=GetLastDrive+#0;
00181:       ret:=GetDriveType( lastDrvP );
00182:       if ( ret=DRIVE_REMOTE ) then DisconnectDrive( lastDrv );
00183:     until ( ret<>DRIVE_REMOTE );
00184:   end;
00185:   //-----------------------------------------
00186:   procedure TFNetDrv.FormCreate(Sender: TObject);
00187:   begin
00188:     InitVar;
00189:   end;
00190:   procedure TFNetDrv.FormCloseQuery(Sender: TObject; var CanClose: Boolean);
00191:   begin
00192:     ReleaseVar;
00193:     canClose:=true;
00194:   end;
00195:   procedure TFNetDrv.Button2Click(Sender: TObject);
00196:   begin
00197:     if ( Connect2RemoteSrvr( Edit2.text, Edit1.text, CBDrvList.text ) ) then
00198:     begin
00199:       DirectoryOutline1.drive:=CBDrvList.text[1];
00200:       DirectoryOutline1.update;
00201:       AddDrv2DrvList;
00202:     end;
00203:     DisconnectDrive( CBDrvList.text );
00204:   end;
00205:   procedure TFNetDrv.DirectoryOutline1MouseDown(Sender: TObject;
00206:     Button: TMouseButton; Shift: TShiftState; X, Y: Integer);
00207:   begin
00208:     if ( Button=mbRight ) then
00209:     begin
00210:       fileList.clear;
00211:       GetAllFileLists( DirectoryOutline1.Directory, fileList );
00212:       RichEdit1.lines:=fileList;
00213:     end;
00214:   end;
00215:   procedure TFNetDrv.Splitter1Moved(Sender: TObject);
00216:   begin
00217:     label2.left:=Splitter1.left+5;
00218:   end;
00219:   end.
```

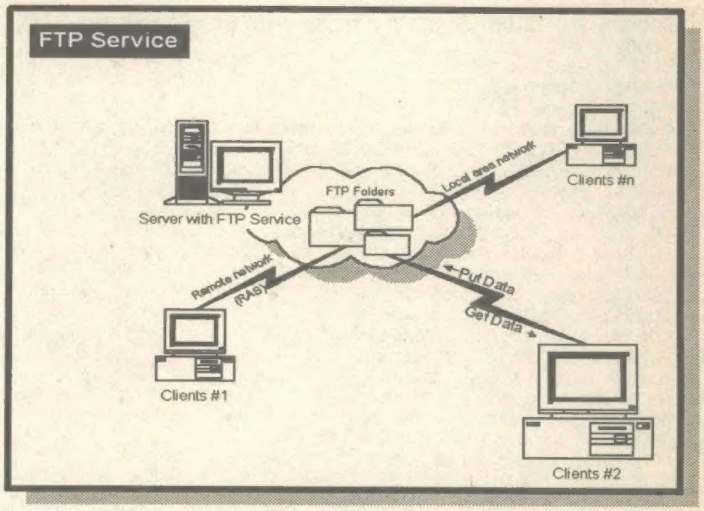
■

ลิสติงที่ 1.2

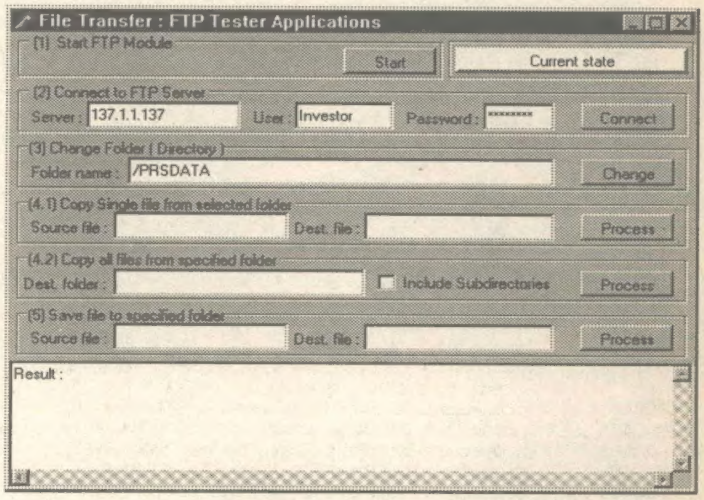
```
object FNetDrv: TFNetDrv
  Left = 193
  Top = 118
  Width = 599
  Height = 388
  Caption = 'File Transfer :: Net Drive'
  Font.Charset = DEFAULT_CHARSET
  Font.Color = clWindowText
  Font.Height = -10
  Font.Name = 'MS Sans Serif'
  Font.Style = []
  OnCloseQuery = FormCloseQuery
  OnCreate = FormCreate
  PixelsPerInch = 96
  TextHeight = 14
  object Splitter1: TSplitter
    Left = 276
    Top = 56
    Width = 3
    Height = 305
    Cursor = crHSplit
    OnMoved = Splitter1Moved
  end
  object Panel5: TPanel
    Left = 0
    Top = 0
    Width = 591
    Height = 56
    Align = alTop
    TabOrder = 0
    object Label1: TLabel
      Left = 10
      Top = 35
      Width = 177
      Height = 14
      Caption = 'Double click to get currect directory :'
      Transparent = True
    end
    object Label2: TLabel
      Left = 280
      Top = 35
      Width = 181
      Height = 14
      Caption = 'Right click to scan current directories :'
      Transparent = True
    end
    object Edit2: TEdit
      Left = 85
      Top = 5
      Width = 226
      Height = 22
      TabOrder = 0
      Text = '\\IPRO2SERV\C'
    end
    object Panel3: TPanel
      Left = 5
      Top = 5
      Width = 76
      Height = 26
      Caption = 'Connect to :'
      Color = clNavy
      Font.Charset = DEFAULT_CHARSET
      Font.Color = clWhite
      Font.Height = -11
      Font.Name = 'MS Sans Serif'
      Font.Style = []
      ParentFont = False
      TabOrder = 1
    end
object Panel4: Tpanel
      Left = 430
      Top = 5
      Width = 56
      Height = 26
      Caption = 'As drive :'
      Color = clNavy
      Font.Charset = DEFAULT_CHARSET
      Font.Color = clWhite
      Font.Height = -11
      Font.Name = 'MS Sans Serif'
      Font.Style = []
      ParentFont = False
      TabOrder = 2
    end
    object CBDrvList: TComboBox
      Left = 490
      Top = 5
      Width = 41
      Height = 22
      ItemHeight = 14
      TabOrder = 3
    end
    object Button2: TButton
      Left = 535
      Top = 5
      Width = 51
      Height = 26
      Caption = 'Connect'
      TabOrder = 4
      OnClick = Button2Click
    end
    object Panel1: TPanel
      Left = 315
      Top = 5
      Width = 56
      Height = 26
      Caption = 'Password'
      Color = clNavy
      Font.Charset = DEFAULT_CHARSET
      Font.Color = clWhite
      Font.Height = -11
      Font.Name = 'MS Sans Serif'
      Font.Style = []
      ParentFont = False
      TabOrder = 5
    end
    object Edit1: TEdit
      Left = 375
      Top = 5
      Width = 51
      Height = 22
      PasswordChar = '*'
      TabOrder = 6
      Text = 'invpro2'
    end
    object CheckBox1: TCheckBox
      Left = 495
      Top = 40
      Width = 96
      Height = 11
      Caption = 'Scan all sub-dir.'
      State = cbChecked
      TabOrder = 7
    end
  end
object DirectoryOutline1: TDirectoryOutline
    Left = 0
    Top = 56
    Width = 276
    Height = 305
    Align = alLeft
    ItemHeight = 13
    OnMouseDown = DirectoryOutline1MouseDown
    Options = [ooDrawFocusRect]
    PictureLeaf.Data = {
      BE010000424DBE010000000000003600
```

ลิสติงที่ 1.2 (ต่อ)

```
        0000280000000E0000000E0000000100
        1000000000008801000000000000000
        00000000000000000000F3C0F3C0F3C
        0F3C0F3C0F3C0F3C0F3C0F3C0F3C0F3C
        0F3C0F3C0F3C0F3C0F3C0F3C0F3C0F3C
        0F3C0F3C0F3C0F3C0F3C0F3C0F3C0F3C
        0F3C0F3C0F3C0F3C0F3C0F3C0F3C0F3C
        0F3C0F3C0F3C0F3C0F3C0F3C0F3C0F3C
        00000000000000000000000000000000
        000000000F3C0F3C0F3C0000E07FFF7F
        E07FFF7FE07FFF7FE07FFF7FE07F0000
        0F3C0F3C0F3C0000FF7FE07FFF7FE07F
        FF7FE07FFF7FE07FFF7F00000F3C0F3C
        0F3C0000E07FFF7FE07FFF7FE07FFF7F
        E07FFF7FE07F00000F3C0F3C0F3C0000
        FF7FE07FFF7FE07FFF7FE07FFF7FE07F
        FF7F00000F3C0F3C0F3C0000E07FFF7F
        E07FFF7FE07FFF7FE07FFF7FE07F0000
        0F3C0F3C0F3C0000FF7FE07FFF7FE07F
        FF7FE07FFF7FE07FFF7F00000F3C0F3C
        0F3C000000000000000000000000000000
        0000000000000F3C0F3C0F3C0F3C0F3C
        0000FF7FE07FFF7FE07F00000F3C0F3C
        0F3C0F3C0F3C0F3C0F3C0F3CEF3D0000
        000000000000EF3D0F3C0F3C0F3C0F3C
        0F3C0F3C0F3C0F3C0F3C0F3C0F3C0F3C
        0F3C0F3C0F3C0F3C0F3C0F3C0F3C0F3C
        0F3C}
    TabOrder = 1
    TextCase = tcAsIs
    Data = {10}
  end
  object RichEdit1: TRichEdit
    Left = 279
    Top = 56
    Width = 312
    Height = 305
    Align = alClient
    ScrollBars = ssBoth
    TabOrder = 2
    WordWrap = False
  end
end
```



รูปที่ 2



รูปที่ 3

File Transfer

FTP (File Transfer Protocol)

FTP หรือ Internet File Transfer Protocol เป็นวิธีการโอนย้ายแฟ้มข้อมูลระหว่าง Remote Host และเครื่อง Clients (Local Machine) ซึ่งเป็นที่นิยมใช้กันอย่างมากในระบบเครือข่าย Internet การใช้ FTP จำเป็นที่จะต้องมี Server ที่เปิดบริการ FTP อาจจะเป็นระบบ Windows NT 4.0 ที่ติดตั้ง Microsoft Internet Information Server ลักษณะการใช้งานคือ เครื่อง Clients ที่ต้องการใช้งานต้องเป็นสมาชิก คือมี User ID และได้รับสิทธิ (Grant permission) ให้สามารถเข้าใช้บริการ FTP ได้ เมื่อได้รับสิทธิแล้ว Clients ก็จะสามารถใช้บริการต่าง ๆ ของ FTP (เช่น Load แฟ้มข้อมูล (Get file), Save แฟ้มข้อมูล (Put file)) ผ่าน Network Device ต่าง ๆ เช่น LAN (Local Area Network), Remote Network ใช้ Modem ติดต่อผ่าน RAS Service) ดังรูปที่ 2

โปรแกรม FTPTstr (FTP Tester Application) แสดงอยู่ในรูปที่ 3 เป็นโปรแกรมแสดงการขอใช้บริการ FTP กับ Server สำหรับโปรแกรมนี้พัฒนาขึ้นด้วย Delphi 3.0 และทดลองใช้บริการ FTP กับ Windows NT 4.0 ที่ติดตั้ง Microsoft Internet Information Server แล้ว (รวมทั้งทดสอบกับเครื่อง RISC6000 และ STRATUS ด้วย) ปรากฏว่าสามารถทำงานได้เป็นอย่างดี

ความต้องการของระบบ

การใช้งานในระบบ FTP นอกจากต้องการ Server เพื่อใช้ติดต่อด้วยแล้ว ในส่วน Clients (ตัวเครื่องที่ติดตั้งโปรแกรม FTP นี้) ต้องรันภายใต้ระบบ Windows 95 หรือ Windows NT (ได้ทั้ง Server และ Work Station) และติดตั้ง Microsoft TCP/IP Protocol

ลิสติงที่ 2.1

```
00001:  // Listing 2.1 : FTPTstr ( UftpTstr.pas )
00002:  //
00003:  unit uFTPTstr;
00004:  interface
00005:  uses
00006:    Windows, Messages, SysUtils, Classes, Graphics,
          Controls, Forms, Dialogs,
00007:    StdCtrls, Buttons, uFTPMain, ExtCtrls;
00008:  type
00009:    TFtpTstr = class(TForm)
00010:      GroupBox1: TGroupBox;
00011:      BitBtn1: TBitBtn;
00012:      GroupBox2: TGroupBox;
00013:      Edit1: TEdit;
00014:      Label1: TLabel;
00015:      Label2: TLabel;
00016:      Edit2: TEdit;
00017:      Label3: TLabel;
00018:      Edit3: TEdit;
00019:      BitBtn2: TBitBtn;
00020:      GroupBox3: TGroupBox;
00021:      Label4: TLabel;
00022:      Edit4: TEdit;
00023:      BitBtn3: TBitBtn;
00024:      GroupBox4: TGroupBox;
00025:      Label5: TLabel;
00026:      Edit5: TEdit;
00027:      Label6: TLabel;
00028:      Edit6: TEdit;
00029:      BitBtn4: TBitBtn;
00030:      GroupBox5: TGroupBox;
00031:      Label7: TLabel;
00032:      Edit7: TEdit;
00033:      CheckBox1: TCheckBox;
00034:      BitBtn5: TBitBtn;
00035:      Memo1: TMemo;
00036:      GroupBox6: TGroupBox;
00037:      Label8: TLabel;
00038:      Edit8: TEdit;
00039:      BitBtn6: TBitBtn;
00040:      Label9: TLabel;
00041:      Edit9: TEdit;
00042:      Panel1: TPanel;
00043:      Bevel1: TBevel;
00044:      procedure FormCreate(Sender: TObject);
00045:      procedure FormCloseQuery(Sender: TObject;
              var CanClose: Boolean);
00046:      procedure BitBtn1Click(Sender: TObject);
00047:      procedure BitBtn2Click(Sender: TObject);
00048:      procedure BitBtn3Click(Sender: TObject);
00049:      procedure BitBtn4Click(Sender: TObject);
00050:      procedure BitBtn5Click(Sender: TObject);
00051:      procedure BitBtn6Click(Sender: TObject);
00052:    private
00053:      { Private declarations }
00054:      procedure InitVar;
00055:      procedure ReleaseVar;
00056:      procedure Echo( content : string );
00057:    public
00058:      { Public declarations }
00059:    protected
00060:      { Protected declarations }
00061:      procedure WmStateChanged( var mmsg : TMessage );
            message wm_state_changed;
00062:      procedure WmLoadCompleted( var mmsg : TMessage );
            message wm_load_completed;
00063:      procedure WmLoadFailded( var mmsg : TMessage );
            message wm_load_failed;
00064:      procedure WmSaveCompleted( var mmsg : TMessage );
            message wm_save_completed;
00065:      procedure WmSaveFailed( var mmsg : TMessage );
            message wm_save_failed;
00066:      procedure WmFtpSuccess( var mmsg : TMessage );
            message wm_ftp_success;
00067:    end;
00068:  var
00069:    FftpTstr: TFftpTstr;
00070:  implementation
00071:  {$R *.DFM}
00072:  //-------------------------------------------------
00073:  procedure TFftpTstr.InitVar;
00074:  begin
00075:  // Initialized part
00076:  end;
00077:  procedure TFftpTstr.ReleaseVar;
00078:  begin
00079:  // Release part
00080:  end;
00081:  procedure TFftpTstr.Echo( content : string );
00082:  begin
00083:    memo1.lines.add( content );
00084:  end;
00085:  //-------------------------------------------------
00086:  procedure TFftpTstr.WmStateChanged
        ( var mmsg : TMessage );
00087:  var  p : pchar; s : string;
00088:  begin
00089:    // mmsg.wparam = keep state value
00090:    p:=pchar( mmsg.lparam ); s:=StrPas( p );
00091:    Panel1.caption:=s;
00092:  end;
00093:  procedure TFftpTstr.WmLoadCompleted
        ( var mmsg : TMessage );
00094:  var  p : pchar; s : string;
00095:  begin
00096:    // mmsg.wparam = keep process state value
00097:    p:=pchar( mmsg.lparam ); s:=StrPas( p );
00098:    Echo( s+', load completed.' );
00099:  end;
00100:  procedure TFftpTstr.WmLoadFailded
        ( var mmsg : TMessage );
00101:  var  p : pchar; s : string;
00102:  begin
00103:    // mmsg.wparam = keep process state value
00104:    p:=pchar( mmsg.lparam ); s:=StrPas( p );
00105:    Echo( s+', load failed.' );
00106:  end;
00107:  procedure TFftpTstr.WmSaveCompleted
        ( var mmsg : TMessage );
00108:  var  p : pchar; s : string;
00109:  begin
00110:    // mmsg.wparam = keep process state value
00111:    p:=pchar( mmsg.lparam ); s:=StrPas( p );
00112:    Echo( s+', save completed.' );
00113:  end;
00114:  procedure TFftpTstr.WmSaveFailed
        ( var mmsg : TMessage );
00115:  var  p : pchar; s : string;
00116:  begin
00117:    // mmsg.wparam = keep process state value
00118:    p:=pchar( mmsg.lparam ); s:=StrPas( p );
00119:    Echo( s+', save failed.' );
00120:  end;
00121:  procedure TFftpTstr.WmFtpSuccess
        ( var mmsg : TMessage );
00122:  begin
00123:    // get success message here..
00124:  end;
00125:  //-------------------------------------------------
00126:  procedure TFftpTstr.FormCreate(Sender: TObject);
00127:  begin
00128:    InitVar;
00129:  end;
00130:  procedure TFftpTstr.FormCloseQuery
        (Sender: TObject; var CanClose: Boolean);
00131:  begin
00132:    ReleaseVar;
00133:    canclose:=true;
00134:  end;
00135:  procedure TFftpTstr.BitBtn1Click(Sender: TObject)
00136:  begin
00137:    if ( FftpMain=nil ) then
00138:      Application.CreateForm( TFftpMain, FftpMain )
00139:    FftpMain.FTPInit( FftpTstr.Handle, 21, false );
00140:  end;
00141:  procedure TFftpTstr.BitBtn2Click(Sender: TObject)
00142:  begin
00143:    FftpMain.FTPConnect( Edit1.text, Edit2.text, Edit3.text );
00144:  end;
```

ลิสติงที่ 2.1 (ต่อ)

```
00145:  procedure TFftpTstr.BitBtn3Click(Sender: TObject);
00146:  begin
00147:    FftpMain.FTPSetCurrentFolder( Edit4.text );
00148:  end;
00149:  procedure TFftpTstr.BitBtn4Click(Sender: TObject);
00150:  begin
00151:    FftpMain.FTPLoadFIle( Edit5.text, Edit6.text );
00152:  end;
00153:  procedure TFftpTstr.BitBtn5Click(Sender: TObject);
00154:  begin
00155:    FftpMain.FTPLoadFolder( Edit7.text, CheckBox1.checked );
00156:  end;
00157:  procedure TFftpTstr.BitBtn6Click(Sender: TObject);
00158:  begin
00159:    FftpMain.FTPSaveFile( Edit8.text, Edit9.text );
00160:  end;
00161:  end.
```

ลิสติงที่ 2.2

```
object FftpTstr: TFftpTstr
  Left = 201
  Top = 108
  BorderStyle = bsSingle
  Caption = 'File Transfer : FTP Tester Applications'
  ClientHeight = 368
  ClientWidth = 491
  Font.Charset = DEFAULT_CHARSET
  Font.Color = clWindowText
  Font.Height = -11
  Font.Name = 'MS Sans Serif'
  Font.Style = []
  OnCloseQuery = FormCloseQuery
  OnCreate = FormCreate
  PixelsPerInch = 96
  TextHeight = 14
  object Bevel1: TBevel
    Left = 315
    Top = 5
    Width = 176
    Height = 31
    Shape = bsFrame
  end
  object GroupBox1: TGroupBox
    Left = 5
    Top = 0
    Width = 306
    Height = 36
    Caption = ' (1)  Start FTP Module '
    TabOrder = 0
    object BitBtn1: TBitBtn
      Left = 235
      Top = 11
      Width = 66
      Height = 21
      Caption = 'Start'
      TabOrder = 0
      OnClick = BitBtn1Click
    end
  end
  object GroupBox2: TGroupBox
    Left = 5
    Top = 40
    Width = 481
    Height = 41
    Caption = ' (2) Connect to FTP Server '
    TabOrder = 1
    object Label1: TLabel
      Left = 10
      Top = 20
      Width = 37
      Height = 14
      Caption = 'Server :'
    end
    object Label2: TLabel
      Left = 170
      Top = 20
      Width = 28
      Height = 14
      Caption = 'User :'
    end
    object Label3: TLabel
      Left = 280
      Top = 20
      Width = 52
      Height = 14
      Caption = 'Password :'
    end
    object Edit1: TEdit
      Left = 50
      Top = 15
      Width = 111
      Height = 22
      TabOrder = 0
      Text = '137.1.1.137'
    end
    object Edit2: TEdit
      Left = 200
      Top = 15
      Width = 71
      Height = 22
      TabOrder = 1
      Text = 'Investor'
    end
    object Edit3: TEdit
      Left = 335
      Top = 15
      Width = 51
      Height = 22
      PasswordChar = '*'
      TabOrder = 2
      Text = 'password'
    end
    object BitBtn2: TBitBtn
      Left = 405
      Top = 15
      Width = 66
      Height = 21
      Caption = 'Connect'
      TabOrder = 3
      OnClick = BitBtn2Click
    end
  end
  object GroupBox3: TGroupBox
    Left = 5
    Top = 85
    Width = 481
    Height = 41
    Caption = '(3) Change Folder ( Directory )'
    TabOrder = 2
    object Label4: TLabel
      Left = 10
      Top = 20
      Width = 64
      Height = 14
      Caption = 'Folder name :'
    end
    object Edit4: TEdit
      Left = 80
      Top = 15
      Width = 306
      Height = 22
      TabOrder = 0
      Text = '/PRSDATA'
    end
    object BitBtn3: TBitBtn
      Left = 405
      Top = 15
      Width = 66
      Height = 21
      Caption = 'Change'
      TabOrder = 1
      OnClick = BitBtn3Click
    end
```

ลิสติงที่ 2.2 (ต่อ)

```
  end
  object GroupBox4: TGroupBox
    Left = 5
    Top = 130
    Width = 481
    Height = 41
    Caption = '(4.1) Copy Single file from selected folder '
    TabOrder = 3
    object Label5: TLabel
      Left = 10
      Top = 19
      Width = 56
      Height = 14
      Caption = 'Source file :'
    end
object Label6: TLabel
      Left = 200
      Top = 19
      Width = 47
      Height = 14
      Caption = 'Dest. file :'
    end
    object Edit5: TEdit
      Left = 70
      Top = 14
      Width = 126
      Height = 22
      TabOrder = 0
    end
    object Edit6: TEdit
      Left = 250
      Top = 14
      Width = 136
      Height = 22
      TabOrder = 1
    end
    object BitBtn4: TBitBtn
      Left = 405
      Top = 14
      Width = 66
      Height = 21
      Caption = 'Process'
      TabOrder = 2
      OnClick = BitBtn4Click
    end
  end
  object GroupBox5: TGroupBox
    Left = 5
    Top = 175
    Width = 481
    Height = 41
    Caption = '(4.2) Copy all files from specified folder '
    TabOrder = 4
    object Label7: TLabel
      Left = 5
      Top = 20
      Width = 60
      Height = 14
      Caption = 'Dest. folder :'
    end
    object Edit7: TEdit
      Left = 70
      Top = 15
      Width = 181
      Height = 22
      TabOrder = 0
    end
    object CheckBox1: TCheckBox
      Left = 260
      Top = 15
      Width = 126
      Height = 21
      Caption = 'Include Subdirectories'
      TabOrder = 1
    end
    object BitBtn5: TBitBtn
      Left = 405
      Top = 15
      Width = 66
      Height = 21
      Caption = 'Process'
      TabOrder = 2
      OnClick = BitBtn5Click
    end
  end
object Memo1: TMemo
    Left = 0
    Top = 265
    Width = 491
    Height = 103
    Align = alBottom
    Lines.Strings = (
      'Result :')
    ScrollBars = ssBoth
    TabOrder = 5
  end
  object GroupBox6: TGroupBox
    Left = 5
    Top = 220
    Width = 481
    Height = 41
    Caption = '(5) Save file to specified folder'
    TabOrder = 6
    object Label8: TLabel
      Left = 10
      Top = 19
      Width = 56
      Height = 14
      Caption = 'Source file :'
    end
    object Label9: TLabel
      Left = 200
      Top = 19
      Width = 47
      Height = 14
      Caption = 'Dest. file :'
    end
    object Edit8: TEdit
      Left = 70
      Top = 14
      Width = 126
      Height = 22
      TabOrder = 0
    end
    object BitBtn6: TBitBtn
      Left = 405
      Top = 14
      Width = 66
      Height = 21
      Caption = 'Process'
      TabOrder = 1
      OnClick = BitBtn6Click
    end
    object Edit9: TEdit
      Left = 250
      Top = 14
      Width = 136
      Height = 22
      TabOrder = 2
    end
  end
  object Panel1: TPanel
    Left = 320
    Top = 10
    Width = 166
    Height = 21
    BevelWidth = 3
    Caption = 'Current state'
    Color = clWhite
    Font.Charset = DEFAULT_CHARSET
    Font.Color = clWindowText
    Font.Height = -11
    Font.Name = 'MS Sans Serif'
    Font.Style = []
    ParentFont = False
    TabOrder = 7
  end
end
```

การติดตั้ง Microsoft TCP/IP Protocol บน Windows NT อาจต้องติดต่อกับ Administrator เพื่อให้ทำการติดตั้งให้ สำหรับ Windows 95 สามารถทำเองได้โดยคลิกขวาที่ไอคอน Network Neighborhood เลือกเมนู Properties เพื่อเข้าสู่หน้าต่าง Network กดปุ่ม Add เลือกรายการ Protocol เข้าสู่หน้าต่าง Select Network Protocol แล้วเลือกรายการ Manufacturers เป็น "Microsoft" และเลือกรายการ Network Protocols เป็น "TCP/IP"

จากการที่ต้องการออกแบบโปรแกรมให้สามารถนำโปรแกรมในส่วนของ FTP ไปใช้งานได้เลยโดยไม่จำเป็นต้องเขียนใหม่ ผมจึงได้แบ่งโปรแกรมออกเป็น 2 ส่วน คือส่วนที่เป็นหน้าต่างหลัก (FTPTstr : uFTPTstr.pas) ใช้เป็นโปรแกรมทดสอบการทำงาน แสดงอยู่ในลิสติงที่ 2.1 และชุด Components ที่เกี่ยวข้องแสดงอยู่ในลิสติงที่ 2.2

สร้างโปรแกรม FTPTstr

สำหรับการสร้างโปรแกรม FTPTstr ให้สร้าง Form ตามลิสติงที่ 2.1 และ 2.2 ให้สังเกตบรรทัดที่ 59-66 เป็นการเพิ่มส่วน Protected Declaration คือ

```
protected
    { Protected declarations }
    procedure WmStateChanged( var mmsg : TMessage );
    message wm_state_changed;
    procedure WmLoadCompleted( var mmsg : TMessage );
    message wm_load_completed;
    procedure WmLoadFailded( var mmsg : TMessage );
    message wm_load_failed;
    procedure WmSaveCompleted( var mmsg : TMessage );
    message wm_save_completed;
    procedure WmSaveFailed( var mmsg : TMessage );
    message wm_save_failed;
    procedure WmFtpSuccess( var mmsg : TMessage );
    message wm_ftp_success;
```

กิจกรรมทั้งหมดนี้เป็นกิจกรรมดักรับเมสเสจที่หน้าต่าง FTP ส่งมาให้ โดย

กิจกรรม WmStateChanged เป็นกิจกรรมสำหรับดักรับเมสเสจ wm_state_changed ใช้ดักรับ (Monitor) การทำงานใน State ระดับต่าง ๆ ของหน้าต่าง FTP เช่น Connecting, Disconnected ค่าที่ส่งให้มี 2 ค่าคือ mmsg.wparam เป็นค่าคงที่ของ State ต่าง ๆ และ mmsg.lparam เป็นค่า long pointer ระบุตำแหน่ง Address ของความหมายในแต่ละ State สามารถแปลงกลับเป็น pointer ด้วยคำสั่ง

```
p:=pChar(mmsg.lparam);//p เป็นตัวแปรชนิด pChar
s:=StrPas(p);//s เป็นตัวแปรชนิด ShortString
```

กิจกรรม WmLoadCompleted เป็นกิจกรรมดักรับเมสเสจ wm_load_completed เมื่อหน้าต่าง FTP ทำการ Load แฟ้มข้อมูลที่ต้องการแล้ว จะส่งชื่อแฟ้มข้อมูลที่ Load สำเร็จมาทาง mmsg.lparam (หมายเหตุ : กิจกรรมนี้ตอบสนองคำสั่งเปลี่ยน Folder ด้วย โดยจะรับค่า null แทน)

กิจกรรม WmLoadFailed ดักรับเมสเสจแจ้งการเกิดปัญหาการ Load แฟ้มข้อมูล จะส่งชื่อแฟ้มข้อมูลที่ Load ไม่สำเร็จ ทาง mmsg.lparam

กิจกรรม WmSaveCompleted ดักรับเมสเสจการ Save สำเร็จ ส่งชื่อแฟ้มข้อมูลที่ทำการ Save มาทาง mmsg.lparam

กิจกรรม WmSaveFailed ดักรับเมสเสจการเกิดปัญหาการ Save แฟ้มข้อมูลที่ Host จะส่งชื่อแฟ้มข้อมูลที่มีปัญหามาทาง mmsg.lparam

กิจกรรม WmFtpSuccess ดักรับเมสเสจการแจ้งการประมวลผลสำเร็จ (Load single file, Load Folder, Save file) ผ่านค่าชนิดการประมวลผลมาทาง mmsg.wparam

กิจกรรมที่สำคัญสำหรับโปรแกรมทดสอบหน้าต่าง FTP มีดังนี้

กิจกรรม TFftpTstr.BitBtn1Click(Sender: TObject); เป็นกิจกรรมสำหรับสร้างหน้าต่าง FTP ด้วยคำสั่ง Application.CreateForm และ Initialized หน้าต่าง FTP ด้วยกิจกรรมชนิด Public ของหน้าต่าง FTP คือ FTPInit โดยผ่านค่า Handle ของหน้าต่างหลัก, หมายเลข Port ที่ต้องการติดต่อ (ในที่นี้เป็น 21) และ DebugFlg กำหนดให้เป็น false

กิจกรรม TFftpTstr.BitBtn2Click(Sender: TObject); กิจกรรมที่สั่งให้ FTP Object ติดต่อไปยังเครื่อง Host Computer จะระบุชื่อหมายเลข IP ของ Host, ชื่อ User และหมายเลข Password

กิจกรรม TFftpTstr.BitBtn3Click(Sender: TObject); กิจกรรมสำหรับสั่งให้หน้าต่าง FTP เปลี่ยนการติดต่อกับ Folder ปัจจุบันของ Host เป็น Folder ใหม่ที่ระบุใน Edit4.text

กิจกรรม TFftpTstr.BitBtn4Click(Sender: TObject); กิจกรรมสั่งให้หน้าต่าง FTP ทำการ Load แฟ้มข้อมูลที่ระบุอยู่ใน Edit5.text (จาก Host) และบันทึกลงเป็นแฟ้มข้อมูล Edit6.text ที่ตัว Clients เอง

กิจกรรม TFftpTstr.BitBtn5Click(Sender: TObject); กิจกรรมสำหรับคัดลอกแฟ้มข้อมูลทั้งหมดที่อยู่ใน Folder ปัจจุบันของ Host มาไว้ที่ Edit7.text โดยถ้ากำหนดค่า CheckBox1.checked เป็น TRUE จะเป็นการ Scan

กิจกรรม TFftpTstr.BitBtn6Click(Sender: TObject); กิจกรรมสำหรับบันทึกแฟ้มข้อมูลที่ระบุใน Edit8.text จาก Clients ไปที่ Edit9.text สำหรับ Host

สำหรับโปรแกรมหน้าต่าง FTP (uFtpMain.pas) ซึ่งเป็นหัวใจของการใช้งานระบบ FTP ผมขออนุญาตเก็บไว้คุยกันฉบับหน้าครับ เนื่องจากเนื้อหาค่อนข้างมาก พบกันใหม่ฉบับหน้า สวัสดีครับ